Soft-Input Soft-Output Sphere Decoding

Christoph Studer
Integrated Systems Laboratory
ETH Zurich, 8092 Zurich, Switzerland
Email: studer@iis.ee.ethz.ch

Helmut Bölcskei
Communication Technology Laboratory
ETH Zurich, 8092 Zurich, Switzerland
Email: boelcskei@nari.ee.ethz.ch

***Abstract*—Soft-input soft-output (SISO) detection algorithms form the basis for iterative decoding. The associated computational complexity often poses significant challenges for practical receiver implementations, in particular in the context of multiple-input multiple-output wireless systems. In this paper, we present a low-complexity SISO sphere decoder which is based on the single tree search paradigm, proposed originally for soft-output detection in Studer *et al.*, *IEEE J-SAC*, 2008. The algorithm incorporates clipping of the extrinsic log-likelihood ratios in the tree search, which not only results in significant complexity savings, but also allows to cover a large performance/complexity trade-off region by adjusting a single parameter.**

I. Introduction

Soft-input soft-output (SISO) detection in multiple-input multiple-output (MIMO) systems constitutes the basis for iterative decoding, which, in general, achieves significantly better performance than decoding based on hard-output or soft-output-only detection algorithms. Unfortunately, this performance gain comes at the cost of a significant, often prohibitive (in terms of practical implementation), increase in terms of computational complexity.

Implementing different algorithms, each optimized for a maximum allowed detection effort or for a particular system configuration, would entail considerable circuit complexity. A practical SISO detector for MIMO systems should therefore not only exhibit low computational complexity but also cover a wide range of easily adjustable performance/complexity trade-offs.

The single tree search (STS) soft-output sphere decoder (SD) in combination with log-likelihood ratio (LLR) clipping [1] has been demonstrated to be suitable for VLSI implementation and is efficiently tunable between max-log optimal soft-output and low-complexity hard-output detection performance. The STS-SD concept is therefore a promising basis for efficient SISO detection in MIMO systems.

Contributions: We describe a SISO STS-SD algorithm that is tunable between max-log optimal SISO and hard-output maximum a posteriori (MAP) detection performance. To this end, we extend the soft-output STS-SD algorithm described in [1] by a max-log optimal a priori information processing method that significantly reduces the tree-search complexity compared to, e.g., [2], [3], and avoids the computation of transcendental functions. The basic idea of the proposed approach is to incorporate clipping of the extrinsic LLRs into the tree search. This requires that the list administration concept and the tree pruning criterion proposed for soft-output STS-SD in [1] be suitably modified. Simulation results show that the SISO STS-SD with extrinsic LLR clipping attains close to max-log optimal SISO performance at remarkably low computational complexity and, in addition, offers a significantly larger performance/complexity trade-off region than the soft-output STS-SD in [1].

Notation: Matrices are set in boldface capital letters, vectors in boldface lowercase letters. The superscripts ^{T} and ^{H} stand for transpose and conjugate transpose, respectively. We write $A_{i,j}$ for the entry in the ith row and jth column of the matrix $\mathbf{A}$ and b_i for the ith entry of the vector $\mathbf{b} = [\, b_1 \; b_2 \; \cdots \; b_N \,]^T$. $\mathbf{I}_N$ denotes the $N \times N$ identity matrix. Slightly abusing common terminology, we call an $N \times M$ matrix $\mathbf{A}$, where $N \geq M$, satisfying $\mathbf{A}^H \mathbf{A} = \mathbf{I}_M$, unitary. $|\mathcal{O}|$ denotes the cardinality of the set $\mathcal{O}$. The probability of an event $\mathcal{Z}$ is denoted by $\mathrm{P}[\mathcal{Z}]$. $\overline{x}$ is the binary complement of $x \in \{+1, -1\}$, i.e., $\overline{x} = -x$.

II. Soft-Input Soft-Output Sphere Decoding

Consider a MIMO system with M_T transmit and $M_R \geq M_T$ receive antennas. The coded bit-stream to be transmitted is mapped to (a sequence of) M_T-dimensional transmit symbol vectors $\mathbf{s} \in \mathcal{O}^{M_T}$, where $\mathcal{O}$ stands for the underlying complex scalar constellation and $|\mathcal{O}| = 2^Q$. Each symbol vector $\mathbf{s}$ is associated with a label vector $\mathbf{x}$ containing $M_T Q$ binary values chosen from the set $\{+1, -1\}$ where the null element (0 in binary logic) of GF(2) corresponds to $+1$. The corresponding bits are denoted by $x_{j,b}$, where the indices j and b refer to the bth bit in the binary label of the jth entry of the symbol vector $\mathbf{s} = [\, s_1 \; s_2 \; \cdots \; s_{M_T} \,]^T$. The associated complex baseband input-output relation is given by

$$\mathbf{y} = \mathbf{H}\mathbf{s} + \mathbf{n} \tag{1}$$

where $\mathbf{H}$ stands for the $M_R \times M_T$ channel matrix, $\mathbf{y}$ is the M_R-dimensional received signal vector, and $\mathbf{n}$ is an i.i.d. circularly symmetric complex Gaussian distributed M_R-dimensional noise vector with variance N_o per complex entry.

A. Max-Log LLR Computation as a Tree Search

SISO detection for MIMO systems requires computation of the LLRs [4], [5]

$$L_{j,b} \triangleq \log\left(\frac{\mathrm{P}[x_{j,b} = +1 | \mathbf{y}, \mathbf{H}]}{\mathrm{P}[x_{j,b} = -1 | \mathbf{y}, \mathbf{H}]}\right) \tag{2}$$

for all bits $j = 1, 2, \ldots, M_T$, $b = 1, 2, \ldots, Q$, in the label $\mathbf{x}$. Transforming (2) into a tree-search problem and using the sphere decoding algorithm allows efficient computation of the LLRs [6], [3], [1]. To this end, the channel matrix $\mathbf{H}$ is first QR-decomposed according to $\mathbf{H} = \mathbf{QR}$, where the

$M_R \times M_T$ matrix $\mathbf{Q}$ is unitary and the $M_T \times M_T$ upper-triangular matrix $\mathbf{R}$ has real-valued positive entries on its main diagonal. Left-multiplying (1) by $\mathbf{Q}^H$ leads to the modified input-output relation $\tilde{\mathbf{y}} = \mathbf{R}\mathbf{s} + \mathbf{Q}^H\mathbf{n}$, where $\tilde{\mathbf{y}} = \mathbf{Q}^H\mathbf{y}$. Noting that $\mathbf{Q}^H\mathbf{n}$ is also i.i.d. circularly symmetric complex Gaussian and using the max-log approximation leads to the *intrinsic* LLRs [4]

$$L^D_{j,b} \triangleq \min_{\mathbf{s}\in\mathcal{X}^{(-1)}_{j,b}} \left\{ \frac{1}{N_o} \left\| \tilde{\mathbf{y}} - \mathbf{R}\mathbf{s} \right\|^2 - \log \mathrm{P}[\mathbf{s}] \right\} - \min_{\mathbf{s}\in\mathcal{X}^{(+1)}_{j,b}} \left\{ \frac{1}{N_o} \left\| \tilde{\mathbf{y}} - \mathbf{R}\mathbf{s} \right\|^2 - \log \mathrm{P}[\mathbf{s}] \right\} \qquad (3)$$

where $\mathcal{X}^{(-1)}_{j,b}$ and $\mathcal{X}^{(+1)}_{j,b}$ are the sets of symbol vectors that have the bit corresponding to the indices j and b equal to -1 and $+1$, respectively. In the following, we consider an iterative MIMO decoder as depicted in Fig. 1. A soft-input soft-output MIMO detector computes intrinsic LLRs according to (3) based on the received signal vector $\mathbf{y}$ and on a priori probabilities in the form of the *a priori* LLRs $L^A_{j,b} \triangleq \log\left(\frac{\mathrm{P}[x_{j,b}=+1]}{\mathrm{P}[x_{j,b}=-1]}\right)$ and delivers the *extrinsic* LLRs

$$L^E_{j,b} = L^D_{j,b} - L^A_{j,b}, \qquad \forall\, j, b, \qquad (4)$$

to a subsequent SISO channel decoder.

For each bit, one of the two minima in (3) corresponds to

$$\lambda^{\mathrm{MAP}} \triangleq \frac{1}{N_o} \left\| \tilde{\mathbf{y}} - \mathbf{R}\mathbf{s}^{\mathrm{MAP}} \right\|^2 - \log \mathrm{P}\left[\mathbf{s}^{\mathrm{MAP}}\right] \qquad (5)$$

which is associated with the MAP solution of the MIMO detection problem

$$\mathbf{s}^{\mathrm{MAP}} = \arg\min_{\mathbf{s}\in\mathcal{O}^{M_T}} \left\{ \frac{1}{N_o} \left\| \tilde{\mathbf{y}} - \mathbf{R}\mathbf{s} \right\|^2 - \log \mathrm{P}[\mathbf{s}] \right\}. \qquad (6)$$

The other minimum in (3) can be computed as

$$\lambda^{\overline{\mathrm{MAP}}}_{j,b} \triangleq \min_{\mathbf{s}\in\mathcal{X}_{j,b}^{\left(\overline{x^{\mathrm{MAP}}_{j,b}}\right)}} \left\{ \frac{1}{N_o} \left\| \tilde{\mathbf{y}} - \mathbf{R}\mathbf{s} \right\|^2 - \log \mathrm{P}[\mathbf{s}] \right\} \qquad (7)$$

where the (bit-wise) *counter-hypothesis* $\overline{x^{\mathrm{MAP}}_{j,b}}$ to the MAP hypothesis denotes the binary complement of the bth bit in the label of the jth entry of $\mathbf{s}^{\mathrm{MAP}}$. With the definitions (5) and (7), the intrinsic max-log LLRs in (3) can be written in compact form as

$$L^D_{j,b} = \begin{cases} \lambda^{\overline{\mathrm{MAP}}}_{j,b} - \lambda^{\mathrm{MAP}}, & x^{\mathrm{MAP}}_{j,b} = +1 \\ \lambda^{\mathrm{MAP}} - \lambda^{\overline{\mathrm{MAP}}}_{j,b}, & x^{\mathrm{MAP}}_{j,b} = -1. \end{cases} \qquad (8)$$

We can therefore conclude that efficient max-log optimal soft-input soft-output MIMO detection reduces to efficiently identifying $\mathbf{s}^{\mathrm{MAP}}$, λ^{MAP}, and $\lambda^{\overline{\mathrm{MAP}}}_{j,b}$ $(\forall j, b)$.

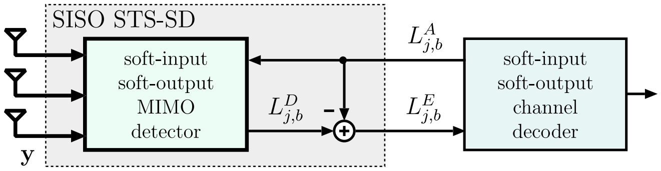
SISO STS-SD
soft-input soft-output MIMO detector
$L^A_{j,b}$
$L^D_{j,b}$
-
$L^E_{j,b}$
soft-input soft-output channel decoder
y

Fig. 1. Iterative MIMO decoder [4]. The SISO STS-SD (corresponding to the dashed box) directly computes extrinsic log-likelihood ratios.

We next define the partial symbol vectors (PSVs) $\mathbf{s}^{(j)} = [\, s_j \; s_{j+1} \; \cdots \; s_{M_T} \,]^T$ and note that they can be arranged in a tree that has its root just above level $j = M_T$ and leaves, on level $j = 1$, which correspond to symbol vectors $\mathbf{s}$. The binary-valued label vector associated with $\mathbf{s}^{(j)}$ will be denoted by $\mathbf{x}^{(j)}$. The distances

$$d(\mathbf{s}) = \frac{1}{N_o} \left\| \tilde{\mathbf{y}} - \mathbf{R}\mathbf{s} \right\|^2 - \log \mathrm{P}[\mathbf{s}]$$

in (5) and (7) can be computed recursively if the individual symbols s_j $(j = 1, 2, \ldots, M_T)$ are statistically independent, i.e., if $\mathrm{P}[\mathbf{s}] = \prod_{j=1}^{M_T} \mathrm{P}[s_j]$. In this case, we have

$$d(\mathbf{s}) = \sum_{j=1}^{M_T} \left(\frac{1}{N_o} \left| \tilde{y}_j - \sum_{i=j}^{M_T} R_{j,i} s_i \right|^2 - \log \mathrm{P}[s_j] \right)$$

which can be evaluated recursively as $d(\mathbf{s}) = d_1$, with the partial distances (PDs)

$$d_j = d_{j+1} + |e_j|, \quad j = M_T, M_T - 1, \ldots, 1,$$

the initialization $d_{M_T+1} = 0$, and the distance increments (DIs)

$$|e_j| = \frac{1}{N_o} \left| \tilde{y}_j - \sum_{i=j}^{M_T} R_{j,i} s_i \right|^2 - \log \mathrm{P}[s_j]\,. \qquad (9)$$

Note that the DIs are non-negative since $-\log \mathrm{P}[s_j] \geq 0$. The dependence of the PDs d_j on the symbol vector $\mathbf{s}$ is only through the PSV $\mathbf{s}^{(j)}$. Thus, the MAP detection problem and the computation of the *intrinsic* max-log LLRs have been transformed into tree-search problems: PSVs and PDs are associated with nodes, branches correspond to DIs. For brevity, we shall often say "the node $\mathbf{s}^{(j)}$" to refer to the node corresponding to the PSV $\mathbf{s}^{(j)}$. We shall furthermore use $d\big(\mathbf{s}^{(j)}\big)$ and $d\big(\mathbf{x}^{(j)}\big)$ interchangeably to denote d_j. Each path from the root node down to a leaf node corresponds to a symbol vector $\mathbf{s} \in \mathcal{O}^{M_T}$. The solution of (5) and (7) corresponds to the leaf associated with the smallest metric in $\mathcal{O}^{M_T}$ and $\mathcal{X}_{j,b}^{\left(\overline{x^{\mathrm{MAP}}_{j,b}}\right)}$, respectively. The SISO STS-SD uses elements of the Schnorr-Euchner SD with radius reduction [7], [8], briefly summarized as follows: The search in the weighted tree is constrained to nodes which lie within a radius r around $\tilde{\mathbf{y}}$ and tree traversal is performed depth-first, visiting the children of a given node in ascending order of their PDs. A node $\mathbf{s}^{(j)}$ with PD d_j can be pruned (along with the entire subtree originating from this node) whenever the pruning criterion $d_j \geq r^2$ is met. In order to avoid the problem of choosing a suitable starting radius, we initialize the algorithm with $r = \infty$ and perform the update $r^2 \leftarrow d(\mathbf{s})$ whenever a valid leaf node $\mathbf{s}$ has been reached. The complexity measure employed in the remainder of the paper corresponds to the number of nodes visited by the decoder including the leaves, but excluding the root.

B. Tree Search for Statistically Independent Bits

Consider the case where all Q bits corresponding to a symbol s_j are statistically independent and the MIMO detector obtains a priori LLRs $L^A_{j,b}$ ($\forall j, b$) from an external device, e.g., a SISO channel decoder as depicted in Fig. 1. We then have [9]

$$\mathrm{P}[s_j] = \prod_{b=1}^{Q} \frac{\exp\left(\frac{1}{2}\left(1+x_{j,b}\right)L^A_{j,b}\right)}{1+\exp\left(L^A_{j,b}\right)}. \tag{10}$$

The contribution of the a priori LLRs to the DIs in (9) can be obtained from (10) as

$$-\log \mathrm{P}[s_j] = -\sum_{b=1}^{Q} \frac{1}{2} x_{j,b} L^A_{j,b} + \tilde{K}_j \tag{11}$$

where the constants

$$\tilde{K}_j = \sum_{b=1}^{Q} \left(\frac{1}{2}\left|L^A_{j,b}\right| + \log\left(1 + \exp\left(-\left|L^A_{j,b}\right|\right)\right)\right) \tag{12}$$

are independent of the binary-valued variables $x_{j,b}$ and $\tilde{K}_j > 0$ for $j = 1, 2, \ldots, M_T$. Because of $-\log \mathrm{P}[s_j] \geq 0$, we can trivially infer from (11) that $-\sum_{b=1}^{Q} \frac{1}{2} x_{j,b} L^A_{j,b} + \tilde{K}_j \geq 0$. From (8) it follows that constant terms (i.e., terms that are independent of the variables $x_{j,b}$ and hence of $\mathbf{s}$) in (5) and (7) cancel out in the computation of the intrinsic LLRs and can therefore be neglected. A straightforward method to avoid the hardware-inefficient task of computing transcendental functions in (12) is to set $\tilde{K}_j = 0$ in the computation of (11). This can, however, lead to branch metrics that are not necessarily non-negative, which would inhibit pruning of the search tree. On the other hand, modifying the DIs in (9) by setting

$$|e_j| \triangleq \frac{1}{N_o} \left| \tilde{y}_j - \sum_{i=j}^{M_T} R_{j,i} s_i \right|^2 - \sum_{b=1}^{Q} \frac{1}{2} x_{j,b} L^A_{j,b} + K_j \tag{13}$$

with $K_j = \sum_{b=1}^{Q} \frac{1}{2}\left|L^A_{j,b}\right|$ also avoids computing transcendental functions while guaranteeing that, since $-x_{j,b} L^A_{j,b} + \left|L^A_{j,b}\right| \geq 0$ $(\forall j, b)$, the so obtained branch metrics are non-negative. Furthermore, as $\tilde{K}_j \geq K_j$, using the modified DIs leads to tighter, but, thanks to (8), still max-log optimal tree pruning, thereby (often significantly) reducing the complexity compared to that obtained through (9).

Note that in [10, Eq. 9], the prior term (11) was approximated as

$$-\log \mathrm{P}[s_j] \approx \sum_{b=1}^{Q} \frac{1}{2} \Big(- x_{j,b} L^A_{j,b} + \left|L^A_{j,b}\right| \Big)$$

for $\left|L^A_{j,b}\right| > 2$ ($b = 1, 2, \ldots, Q$) which corresponds exactly to what was done here to arrive at (13). It is important, though, to realize that using the modified DIs (13) does *not* lead to an approximation of (8), as the neglected $\log(\cdot)$ term in (12) does not affect (8).

III. Extrinsic LLR Computation in a Single Tree Search

Computing the intrinsic max-log LLRs in (8) requires to determine λ^{MAP} and the metrics $\lambda^{\overline{\mathrm{MAP}}}_{j,b}$ associated with the counter-hypotheses. For given j and b the metric $\lambda^{\overline{\mathrm{MAP}}}_{j,b}$ is obtained by traversing only those parts of the search tree that have leaves in $\mathcal{X}_{j,b}^{\left(x_{j,b}^{\overline{\mathrm{MAP}}}\right)}$. The quantities λ^{MAP} and $\lambda^{\overline{\mathrm{MAP}}}_{j,b}$ can be computed using the SD based on the repeated tree search (RTS) approach described in [6]. The RTS strategy results, however, in redundant computations as (often significant) parts of the search tree are revisited during the RTS steps required to determine $\lambda^{\overline{\mathrm{MAP}}}_{j,b}$ ($\forall j, b$). Following the STS paradigm described for soft-output sphere decoding in [1], [3], we note that *efficient* computation of $L^D_{j,b}$ ($\forall j, b$) requires that every node in the tree is visited *at most* once. This can be achieved by searching for the MAP solution and computing the metrics $\lambda^{\overline{\mathrm{MAP}}}_{j,b}$ ($\forall j, b$) concurrently and ensuring that the subtree emanating from a given node in the tree is pruned if it can not lead to an update of either λ^{MAP} or at least one of the $\lambda^{\overline{\mathrm{MAP}}}_{j,b}$. Besides extending the ideas in [1] to take into account a priori information, the main idea underlying the SISO STS-SD presented in this paper is to *directly* compute the extrinsic LLRs $L^E_{j,b}$ through a tree search, rather than computing $L^D_{j,b}$ first and then evaluating (4).

Due to the large dynamic range of LLRs, fixed-point hardware implementations need to constrain the magnitude of the LLR value. Evidently, clipping of the LLR magnitude leads to a degradation in terms of decoder performance. It has been noted in [1], [11] that incorporating LLR clipping into the tree search is very effective in terms of reducing complexity of max-log based soft-output sphere decoding. In addition, as demonstrated in [1], LLR clipping, when built into the tree search also allows to tune the detection algorithm in terms of performance versus complexity by adjusting the LLR clipping level. In the SISO case, we are ultimately interested in the *extrinsic* LLRs $L^E_{j,b}$ and clipping should therefore ensure that $\left|L^E_{j,b}\right| \leq L_{\max}$. It is hence sensible to ask whether clipping of the extrinsic LLRs can be built directly into the tree search. The answer is in the affirmative and the corresponding solution is described below.

To prepare the ground for the formulation of the SISO STS-SD, we write the extrinsic LLRs as

$$L^E_{j,b} = \begin{cases} \Lambda^{\overline{\mathrm{MAP}}}_{j,b} - \lambda^{\mathrm{MAP}}, & x^{\mathrm{MAP}}_{j,b} = +1 \\ \lambda^{\mathrm{MAP}} - \Lambda^{\overline{\mathrm{MAP}}}_{j,b}, & x^{\mathrm{MAP}}_{j,b} = -1 \end{cases} \tag{14}$$

where the quantities

$$\Lambda^{\overline{\mathrm{MAP}}}_{j,b} = \begin{cases} \lambda^{\overline{\mathrm{MAP}}}_{j,b} - L^A_{j,b}, & x^{\mathrm{MAP}}_{j,b} = +1 \\ \lambda^{\overline{\mathrm{MAP}}}_{j,b} + L^A_{j,b}, & x^{\mathrm{MAP}}_{j,b} = -1 \end{cases} \tag{15}$$

will be referred to as the *extrinsic metrics*. For the following developments it will be convenient to define a function $f(\cdot)$ that transforms an intrinsic metric λ with associated a priori LLR L^A and binary label x to an extrinsic metric Λ according to

$$\Lambda = f\left(\lambda, L^A, x\right) = \begin{cases} \lambda - L^A, & x = +1 \\ \lambda + L^A, & x = -1. \end{cases} \tag{16}$$

With this notation, we can rewrite (15) more compactly as $\Lambda^{\overline{\mathrm{MAP}}}_{j,b} = f\left(\lambda^{\overline{\mathrm{MAP}}}_{j,b}, L^A_{j,b}, x^{\mathrm{MAP}}_{j,b}\right)$. The inverse function of (16) transforms an extrinsic metric Λ to an intrinsic metric λ and is defined as

$$\lambda = f^{-1}\left(\Lambda, L^A, x\right) = \begin{cases} \Lambda + L^A, & x = +1 \\ \Lambda - L^A, & x = -1. \end{cases} \tag{17}$$

We emphasize that the tree search algorithm described below produces the extrinsic LLRs $L^{E}_{j,b}$ ($\forall j, b$) in (14) rather than the intrinsic ones in (8). Consequently, careful modification of the list administration steps, the pruning criterion, and the LLR clipping rules of the soft-output algorithm described in [1] is needed.

A. List Administration

The main idea of the STS paradigm is to search the subtree originating from a given node only if the result can lead to an update of either λ^{MAP} or of at least one of the $\Lambda^{\overline{\text{MAP}}}_{j,b}$. To this end, the decoder needs to maintain a list containing the label of the current MAP hypothesis $\mathbf{x}^{\text{MAP}}$, the corresponding metric λ^{MAP}, and all QM_T extrinsic metrics $\Lambda^{\overline{\text{MAP}}}_{j,b}$. The algorithm is initialized with $\lambda^{\text{MAP}} = \Lambda^{\overline{\text{MAP}}}_{j,b} = \infty$ ($\forall\, j, b$). Whenever a leaf with corresponding label $\mathbf{x}$ has been reached, the decoder distinguishes between two cases:

i) MAP Hypothesis Update: If $d(\mathbf{x}) < \lambda^{\text{MAP}}$, a new MAP hypothesis has been found. First, all extrinsic metrics $\Lambda^{\overline{\text{MAP}}}_{j,b}$ for which $x_{j,b} = \overline{x^{\text{MAP}}_{j,b}}$ are updated according to

$$\Lambda^{\overline{\text{MAP}}}_{j,b} \leftarrow f\left(\lambda^{\text{MAP}}, L^{A}_{j,b}, x^{\overline{\text{MAP}}}_{j,b}\right)$$

followed by the updates $\lambda^{\text{MAP}} \leftarrow d(\mathbf{x})$ and $\mathbf{x}^{\text{MAP}} \leftarrow \mathbf{x}$. In other words, for each bit in the MAP hypothesis that is changed in the update process, the metric associated with the *former* MAP hypothesis becomes the extrinsic metric of the *new* counter-hypothesis.

ii) Extrinsic Metric Update: If $d(\mathbf{x}) > \lambda^{\text{MAP}}$, only extrinsic metrics corresponding to counter-hypotheses might be updated. For each $j = 1, 2, \ldots, M_T$, $b = 1, 2, \ldots, Q$ with $x_{j,b} = \overline{x^{\text{MAP}}_{j,b}}$ and $f\left(d(\mathbf{x}), L^{A}_{j,b}, x^{\text{MAP}}_{j,b}\right) < \Lambda^{\overline{\text{MAP}}}_{j,b}$, the SISO STS-SD performs the update

$$\Lambda^{\overline{\text{MAP}}}_{j,b} \leftarrow f\left(d(\mathbf{x}), L^{A}_{j,b}, x^{\text{MAP}}_{j,b}\right). \tag{18}$$

B. Extrinsic LLR Clipping

In order to ensure that the extrinsic LLRs delivered by the algorithm indeed satisfy $\left|L^{E}_{j,b}\right| \leq L_{\max}$ ($\forall j, b$), the following update rule

$$\Lambda^{\overline{\text{MAP}}}_{j,b} \leftarrow \min\left\{\Lambda^{\overline{\text{MAP}}}_{j,b}, \lambda^{\text{MAP}} + L_{\max}\right\}, \quad \forall j, b \tag{19}$$

has to be applied after carrying out the steps in Case i) of the list administration procedure described in Section III-A. We emphasize that for $L_{\max} = \infty$ the decoder attains max-log optimal SISO performance, whereas for $L_{\max} = 0$, the hard-output MAP solution (6) is found.

C. The Pruning Criterion

Consider the node $\mathbf{s}^{(j)}$ on level j corresponding to the label bits $x_{i,b}$ ($i = j, j+1, \ldots, M_T$, $b = 1, 2, \ldots, Q$). Assume that the subtree originating from this node and corresponding to the label bits $x_{i,b}$ ($i = 1, 2, \ldots, j-1$, $b = 1, 2, \ldots, Q$) has not been expanded yet. The criterion for pruning the node $\mathbf{s}^{(j)}$ along with its subtree is compiled from two sets defined as follows:

1) The bits in the partial label $\mathbf{x}^{(j)}$ corresponding to the node $\mathbf{s}^{(j)}$ are compared with the corresponding bits in the label of the current MAP hypothesis. All extrinsic metrics $\Lambda^{\overline{\text{MAP}}}_{j,b}$ with $x_{j,b} = \overline{x^{\text{MAP}}_{j,b}}$ found in this comparison may be affected when searching the subtree originating from $\mathbf{s}^{(j)}$. As the metric $d\big(\mathbf{x}^{(j)}\big)$ is an intrinsic metric, the extrinsic metrics $\Lambda^{\overline{\text{MAP}}}_{j,b}$ need to be mapped to intrinsic metrics according to (17). The resulting set of *intrinsic* metrics, which may be affected by an update, is given by

$$\mathcal{A}_1\left(\mathbf{x}^{(j)}\right) = \left\{ f^{-1}\left(\Lambda^{\overline{\text{MAP}}}_{i,b}, L^{A}_{i,b}, x^{\text{MAP}}_{i,b}\right) \;\middle|\; (i \geq j, \forall b) \right.$$
$$\left. \wedge \left(x_{i,b} = \overline{x^{\text{MAP}}_{i,b}}\right)\right\}.$$

2) The extrinsic metrics $\Lambda^{\overline{\text{MAP}}}_{i,b}$ for $i = 1, 2, \ldots, j-1$, $b = 1, 2, \ldots, Q$ corresponding to the counter-hypotheses in the subtree of $\mathbf{s}^{(j)}$ may be affected as well. Correspondingly, we define

$$\mathcal{A}_2\left(\mathbf{x}^{(j)}\right) = \left\{ f^{-1}\left(\Lambda^{\overline{\text{MAP}}}_{i,b}, L^{A}_{i,b}, x^{\text{MAP}}_{i,b}\right) \;\middle|\; i < j, \forall b \right\}.$$

In summary, the intrinsic metrics which may be affected during the search in the subtree emanating from node $\mathbf{s}^{(j)}$ are given by $\mathcal{A}\big(\mathbf{x}^{(j)}\big) = \{a_l\} = \mathcal{A}_1\big(\mathbf{x}^{(j)}\big) \cup \mathcal{A}_2\big(\mathbf{x}^{(j)}\big)$. The node $\mathbf{s}^{(j)}$ along with its subtree is pruned if the corresponding PD $d\big(\mathbf{x}^{(j)}\big)$ satisfies the pruning criterion

$$d\left(\mathbf{x}^{(j)}\right) > \max_{a_l \in \mathcal{A}\left(\mathbf{x}^{(j)}\right)} a_l.$$

This pruning criterion ensures that a given node and the entire subtree originating from that node are explored only if this could lead to an update of either λ^{MAP} or of at least one of the extrinsic metrics $\Lambda^{\overline{\text{MAP}}}_{j,b}$. Note that λ^{MAP} does not appear in the set $\mathcal{A}\big(\mathbf{x}^{(j)}\big)$ as the update criteria given in Section III-A ensure that λ^{MAP} is always smaller than or equal to all intrinsic metrics associated with the counter-hypotheses.

IV. Simulation Results

Fig. 2 shows performance/complexity trade-off curves[1] for the SISO STS-SD described in Sections II and III. The numbers next to the SISO STS-SD trade-off curves correspond to *normalized* LLR clipping levels given by $L_{\max}N_o$. The average (over channel, noise, and data realizations) complexity corresponds to the *cumulative* tree-search complexity associated with SISO detection over I iterations, where one iteration is defined as using the MIMO detector (and the subsequent channel decoder) once. The curve associated with $I = 1$ hence corresponds to the soft-output STS-SD described in [1]. Increasing the number of iterations allows to reduce the SNR operating point (defined as the minimum SNR required to achieve a frame error rate of 1%) at the cost of increased complexity. We can see that performance improves significantly with increasing number of iterations. Note, however, that for a fixed SNR operating point, the minimum complexity is not

[1] All simulation results are for a convolutionally encoded (rate 1/2, generator polynomials [133_o 171_o], and constraint length 7) MIMO-OFDM system with $M_T = M_R = 4$, 16-QAM symbol constellation with Gray labeling, 64 OFDM tones, a TGn type C channel model [12], and are based on a max-log BCJR channel decoder. One frame consists of 1024 randomly interleaved (across space and frequency) bits corresponding to one (spatial) OFDM symbol. The SNR is per receive antenna.

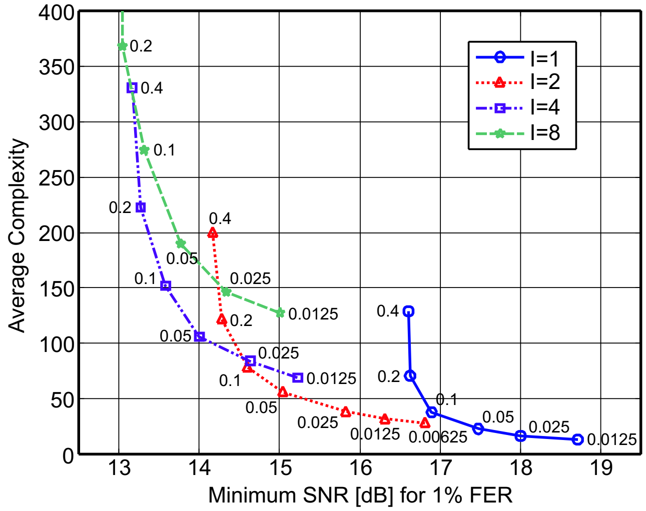

Fig. 2. Performance/complexity trade-off of the SISO STS-SD with sorted QR decomposition (SQRD) as described in [13]. The numbers next to the curves correspond to normalized LLR clipping levels and I denotes the number of iterations over the MIMO detector (and the channel decoder).

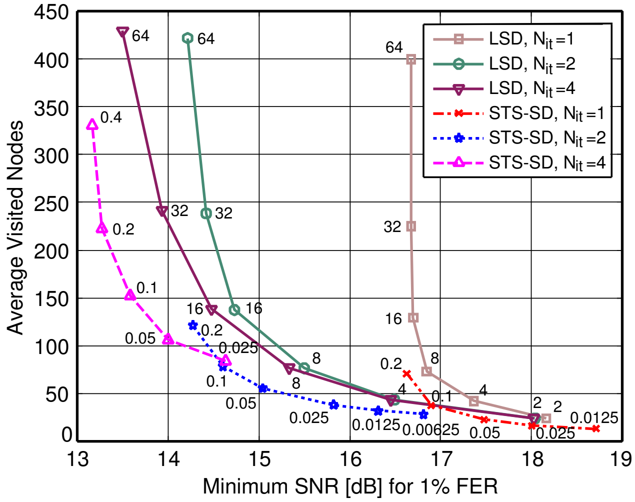

Fig. 3. Performance/complexity (trade-off) comparison of the list sphere decoder (LSD) [4] and the SISO STS-SD (both using SQRD). The numbers next to the curves correspond to the list size for the LSD and to normalized LLR clipping levels for the SISO STS-SD.

necessarily achieved by maximizing the number of iterations I as the trade-off region is parametrized by the LLR clipping level *and* the number of iterations I.

Fig. 3 compares the performance/complexity trade-off achieved by the list sphere decoder (LSD) [4] to that obtained through the SISO STS-SD. For the LSD we take the complexity to equal the number of nodes visited when building the initial candidate list. The (often significant) computational burden incurred by the list administration of the LSD is neglected here. We can draw the following conclusions from Fig. 3:

i) The SISO STS-SD outperforms the LSD for all SNR values.

ii) The LSD requires relatively large list sizes and hence a large amount of memory to approach max-log optimum SISO performance. The underlying reason is that the LSD obtains extrinsic LLRs from a list that has been computed around the maximum-likelihood solution, i.e., in the absence of a priori information. In contrast, the SISO STS-SD requires memory mainly for the extrinsic metrics. The extrinsic LLRs are obtained through a search that is concentrated around the MAP solution. Therefore, the SISO STS-SD tends to require (often significantly) less memory than the LSD.

Besides the LSD, various other SISO detection algorithms for MIMO systems have been developed, see e.g., [5], [10], [14], [15]. For [5], [14] issues indicating potentially prohibitive computational complexity include the requirement for multiple matrix inversions at symbol-vector rate. In contrast, the QR decomposition required for sphere decoding has to be computed only once per frame. The computational complexity of the list-sequential (LISS) algorithm in [10], [15] seems difficult to relate to the complexity measure employed in this paper. However, due to the need for sorting of candidate vectors in a list and the structural similarity of the LISS and the LSD algorithms, we expect their computational complexity behavior to be comparable as well.

References

[1] C. Studer, A. Burg, and H. Bölcskei, “Soft-output sphere decoding: Algorithms and VLSI implementation,” *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 2, pp. 290–300, Feb. 2008.

[2] H. Vikalo and B. Hassibi, “Modified Fincke-Pohst algorithm for low-complexity iterative decoding over multiple antenna channels,” in *Proc. of IEEE International Symposium on Information Theory (ISIT)*, 2002, p. 390.

[3] J. Jaldén and B. Ottersten, “Parallel implementation of a soft output sphere decoder,” in *Proc. of Asilomar Conference on Signals, Systems and Computers*, Nov. 2005, pp. 581–585.

[4] B. M. Hochwald and S. ten Brink, “Achieving near-capacity on a multiple-antenna channel,” *IEEE Trans. on Communications*, vol. 51, no. 3, pp. 389–399, Mar. 2003.

[5] B. Steingrimsson, T. Luo, and K. M. Wong, “Soft quasi-maximum-likelihood detection for multiple-antenna wireless channels,” *IEEE Trans. on Signal Processing*, vol. 51, no. 11, pp. 2710–2719, Nov. 2003.

[6] R.Wang and G. B. Giannakis, “Approaching MIMO channel capacity with reduced-complexity soft sphere decoding,” in *Proc. of IEEE Wireless Communications and Networking Conference (WCNC)*, vol. 3, Mar. 2004, pp. 1620–1625.

[7] E. Agrell, T. Eriksson, A. Vardy, and K. Zeger, “Closest point search in lattices,” *IEEE Trans. on Information Theory*, vol. 48, no. 8, pp. 2201–2214, Aug. 2002.

[8] A. Burg, M. Borgmann, M. Wenk, M. Zellweger, W. Fichtner, and H. Bölcskei, “VLSI implementation of MIMO detection using the sphere decoding algorithm,” *IEEE Journal of Solid-State Circuits*, vol. 40, no. 7, pp. 1566–1577, July 2005.

[9] J. Hagenauer, E. Offer, and L. Papke, “Iterative decoding of binary block and convolutional codes,” *IEEE Trans. on Information Theory*, vol. 42, no. 2, pp. 429–445, Mar. 1996.

[10] S. Bäro, J. Hagenauer, and M. Witzke, “Iterative detection of MIMO transmission using a list-sequential (LISS) detector,” in *Proc. of IEEE International Conference on Communications (ICC)*, vol. 4, May 2003, pp. 2653–2657.

[11] M. S. Yee, “Max-Log-Map sphere decoder,” in *Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 3, Mar. 2005, pp. 1013–1016.

[12] V. Erceg *et al.*, *TGn channel models*, May 2004, IEEE 802.11 document 03/940r4.

[13] D. Wübben, R. Böhnke, J. Rinas, V. Kühn, and K.-D. Kammeyer, “Efficient algorithm for decoding layered space-time codes,” *IEE Electronics Letters*, vol. 37, no. 22, pp. 1348–1350, Oct. 2001.

[14] M. Tüchler, A. C. Singer, and R. Koetter, “Minimum mean squared error equalization using a priori information,” *IEEE Trans. on Signal Processing*, vol. 50, no. 3, pp. 673–983, Mar. 2002.

[15] J. Hagenauer and C. Kuhn, “The list-sequential (LISS) algorithm and its application,” *IEEE Trans. on Communications*, vol. 55, no. 5, pp. 918–928, May 2007.